

OOP Project: Run-Jump Scheduler

A high performance track & field competitions scheduling system

Abstract

The aim of Run-Jump Scheduler project is to develop a scheduling system for small to medium size track and field competitions.

Introduction

Our aim is to revolutionize the area of track and field competitions by introducing advanced AI techniques into competition scheduling! Our goal is to develop a system, which will develop a competition schedule in the best way to allow athletes to participate in multiple events with no hassle and leaving enough time for recovering.

The goal of this pilot project is to develop a scheduling system for domestic and national competitions, to verify scalability of design solutions, proposed by different development teams.

Background and Input Data

System Overview

The system should be able to build a schedule of a typical domestic or national track and field event in the way to allow athletes to participate in all desired disciplines without collisions. Therefore, the primary input data for the scheduler is a list of registered athletes and the capacity of the stadium. The desired software should produce a schedule, from each the start list per station and per athlete can be derived.

The registration list should first be split into gender and age groups. Then, the groups should be also split per discipline. The schedule should be made in the way that there are no collisions, for example if one athlete has to participate in multiple event at the same time, or that two events use the same station. There should be enough time for recovery between competitions per athletes. The system should output a compiled schedule as a single file as well as start lists per age/gender/discipline (they are handed in to the judges at the stations) and list of starts per athlete (given to athletes).

Athletes and Disciplines

The main input data source for the system is the list of registered athletes, along with the disciplines they are willing to compete and their personal records (if available). The list is provided in the form of CSV file with the following columns:

Club	Name	Surname	Sex	Age	Disciplines (all units in running are in m:s.ms and in other disciplines in meters)		
					Running	Jumping	Throwing

					Sprint		Middle		Long	Hurdles	Long	Triple	High	Pole	Shot
					60	200	800	1500	3000	60					
Antarctica IF	Jane	Doe	F	24	6.92	21.87	1:55.82	3:55.17	8:16.60	7.68	7.37	15.36	2.08	5.02	22.50
Antarctica IF	John	Doe	M	22	6.34	19.92	1:42.67	3:31.18	7:24.90	7.30	8.79	17.92	2.43	6.16	22.66
Antarctica IF	Bill	Cody	M	43					9:28:10						31.73

Age Groups

Because of significant performance differences between athletes in different age groups, athletes only compete within their age group. There are multiple age groups depending on the year of birth, summarized in the following table (actual for 2020):

Age Division	Year of Birth
8 & Under	2012+*
9 - 10	2010-2011
11 - 12	2008-2009
13 - 14	2006-2007
15 - 16	2004-2005
17 - 18	2002-2003**

* Per USATF Rule 300.1 (c) "Athletes must be at least seven (7) years of age on December 31 of the current year to compete at the Youth Athletics or Junior Olympic National Championships."

** Athletes who are still 18 through the final day of the USATF National Junior Olympic Track & Field Championships are eligible to compete in the 17-18 age division through that meet. This extended eligibility does not apply to cross country events.

Disciplines

In every discipline, apart from the time to actually perform the trial, extra time is needed to prepare and to leave. Some time is also required for station maintenance. It is required that this time is accounted for in the calculations. The following paragraphs describe the approximate format of competitions in all disciplines.

Running

In running disciplines, the competitions are performed in multiple stages. First, athletes compete in qualifications. After that, best from all qualifications rounds compete in finals. The running disciplines include 60m, 200m, 800m, 1500m, 3000m, and 60m hurdles.

Jumping

In long jumps, every athlete can make up to two trials. The best is counted as a final result. In high jumps and pole vault, athletes start from the same level and can make up to two trials per height. The height is increased after every round by 5 cm. For example, all athletes start from 1.00m. After all of them performed their trials, the height is increased. If one has not succeeded on the current height in the first trial, they can make one more trial. If the second trial was unsuccessful, the athlete has to leave. Competitions continue until there is one

athlete (the winner) left.

Throwing

Every athlete has a right to make four trials. The best of those four is counted as a final result. Only shots are currently being conducted indoors.

Stations

The arena layout includes stations for all disciplines, however some may overlap. For example, running disciplines occupy the same running circle, excepting 60m distance.



For this project, consider the following stations:

- Running Circle (6 tracks)
- Spring Line (8 tracks)
- Long/Triple Jump – I
- Long/Triple Jump – II
- High Jump – I
- High Jump – II
- Pole Vault
- Shot Throwing – I
- Shot Throwing – II
- (optional) Award Ceremony Area

An example of a typical track and field arena can be seen at Tybblelundshallen.

Project Scope

This is a pilot project, in which the scheduling should be done for a small size event taking

place in a single indoor arena.

The final program should be an executable launched from command line. The names of source and result files can be hardcoded. However, it is preferable that the program accepts the filenames as arguments from the command line.

After starting, the program should import data from CSV files, make a schedule, and output it into an output CSV file.

The output data should consist of start lists for each station/age/gender and for every athlete. The compiled start list for the complete competition day should be also made available.

Project Reporting

The final report must be prepared using a provided template. The template will contain the basic structure and tips on writing. Further hints on report writing will be provided in a separate lecture.

The evaluation criteria also include evaluation in the form of periodic commit reviews and peer evaluation. Therefore, special attention must be paid for every group member to make timely contributions to the group progress.

Extra Tasks

Extra tasks are not compulsory to do, but are rather an opportunity and provide an extra merit for the final project evaluation. More additional tasks may be released at the later stages of the project.

Extra Task: Consider the entertainment aspect of the competitions: if the winner is known after the first trials, other athletes would not enjoy the attention of the spectators. This can be addressed, for example, by scheduling the athletes with higher personal records later in time. However, fair distribution of time slots must remain a priority and no athlete can suffer from unfair scheduling due to weaker performance.

Extra Task: Consider slots for meals and refreshments for athletes and spectators in the schedules.

Extra Task: Consider Award Ceremony as a separate station, which also needs to be scheduled.

Extra Task: The technical report can be written using LaTeX document preparation system. This task will require some learning, but will make your life easier on the later stages of the project. If interested, ask for more instructions. Or simply check out Overleaf.

Workplan

The project workplan includes several key milestones and deliverables listed below.

Milestone 1: Design Proposal

The goal of this milestone is to develop an architectural proposal, that depicts all key features of the proposed software. Particularly, it should specify the following:

1. The description of the functional modules and workflow in the form of use case description.
2. The class diagram of the proposed system.
3. The interaction diagram of the system.
4. The critical view of the propose system, evaluated in terms of SOLID design principles.

Deliverable 1.1: A written description of the proposal (in the provided template), published on the group discussion board.

Tentative deadline: December 8, 2020 at the latest if the group expects to get feedback from the customer on the proposed solution.

Upon completing this milestone, you will get access to test dataset and a link to fork a development repository.

Milestone 2: First Implementation

After completing the milestone 1, you switch to system development. This will be done in parallel with some documentation work. Starting from this moment, periodic commits from every group member are essential.

Deliverable 2.1: Software project on Git-E with the basic structure in place.

Deliverable 2.2: Basic structure of the report in place, written parts of introduction, description of problem statement, description of tools, frameworks (if any), third-party libraries.

Tentative deadline: December 14, 2020.

Milestone 3: Getting things done

For this milestone, you have to work on fulfilling the system functionality. This is the right time to finish with new features, get rid of loose ends, and prepare the system for the release.

Deliverable 3.1: Ready software must compile and run, provided as a project on Git-E and bundled with typical usage scenarios (command line arguments, if any).

Deliverable 3.2: Ready report.

Tentative deadline: Friday, December 18, 2020.

Milestone 4: Project presentation

During the presentation, the main project achievements must be highlighted. It is also beneficial to indicate possible direction of future development and discuss possible pros and cons of the proposed solution.

Note: the exact content and agenda for the presentation will be arranged at the later stage.

Evaluation Criteria

The project is performed in groups of 3-5 students. Evaluation is performed on a per-student basis. The following criteria are taken into account when evaluating every student in the group:

- The student should be a co-author of the report. E.g. the name of the student should be on the title page. That means that every co-author should make a sufficient contribution outlined in the IEEE Definition of Authorship:
<https://ieeeauthorcenter.ieee.org/publish-with-ieee/publishing-ethics/definition-of-authorship/>
- Review of commit history on GitLab server. Every group member should make commits in terms of source code and documentation. The commits should be made throughout the project. Therefore, every member should be given a sufficient amount of work to do.